

CONTROL DE PROCESOS EN TIEMPO REAL

Examen Cuatrimestral – Febrero 2004

1. (1.0 pto)

Las siguientes versiones de código intentan obtener una copia de la cadena 2 en la cadena 1. Indicar si son o no correctas, por qué, y proponer en su caso una versión adecuada (una sola para todas).

<p style="text-align: center;">(a)</p> <pre>char cad1[5], cad2[]="HOLA"; cad1=cad2;</pre>	<p style="text-align: center;">(b)</p> <pre>char *cad1, cad2[]="HOLA"; cad1=cad2;</pre>
<p style="text-align: center;">(c)</p> <pre>char cad2[]="HOLA"; char cad1[strlen(cad2)+1]; cad1=cad2;</pre>	<p style="text-align: center;">(d)</p> <pre>char *cad1; cad1="HOLA";</pre>

2. (3.0 pto)

Realizar un **programa** en lenguaje C que recibe 2 parámetros en **línea de comandos**: nombre de archivo y número de líneas en el mismo. En cada línea del archivo se encuentran datos de un polinomio $p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a_1 \cdot x + a_0$, indicados en modo texto con el siguiente formato:

n a_n a_{n-1} ... a₁ a₀

El programa leerá del archivo los datos de todos los polinomios, y una vez obtenidos realizará un bucle en que pide por consola el valor de x , y para el mismo obtiene y escribe el valor de todos los polinomios para dicho x . El bucle se termina cuando algún polinomio tome valor 0 para x .

NOTA: No se puede suponer a priori el grado máximo de los polinomios.

Para calcular x^y , utilizar la función de `<math.h>`:

`double pow(double x, double y);`

3. (1.0 pto)

Realizar una **función** que devuelva una **matriz** de tamaño 5x10, declarada de forma **estática**, rellena con los valores 1..50, de la forma siguiente:

$$M = \begin{bmatrix} 1 & 2 & \dots & 10 \\ 11 & 12 & \dots & 20 \\ \vdots & \vdots & \ddots & \vdots \\ 41 & 42 & \dots & 50 \end{bmatrix}$$

Escribir un **ejemplo** de llamada a la función.

No se puede utilizar el acceso a los elementos de la matriz mediante **índices**: `m[i][j]`, `*(m+...)`, o similares.

CONTROL DE PROCESOS EN TIEMPO REAL

Examen Cuatrimestral – Febrero 2004

4. (1.0 pto) Realizar una **función** que, dada una **lista enlazada** con valores enteros ordenados de mayor a menor, y un nuevo valor entero, inserte el mismo en la posición adecuada de la lista.
5. (1.0 pto) Escribir una posible secuencia de instrucciones de máquina para ejecutar el siguiente bucle:
- ```
int t[10];
int i, suma;
...
for (i=0, suma=0; i!=10; i++)
 suma+=t[i];
```
6. (1.0 pto) Describir los niveles desde los que puede acceder el software a los dispositivos de E/S, indicando las características más importantes de cada nivel en cuanto a: uniformidad de acceso, organización de acceso, capacidad de interactuar con cada dispositivo particular, velocidad de acceso.
7. (1.0 pto) ¿Es siempre necesario un Sistema Operativo de Tiempo Real para realizar el control de un proceso en tiempo real? ¿En qué casos es aconsejable? ¿Cómo se puede realizar el sistema en tiempo real sin el concurso de un R.T.O.S.?
8. (1.0 pto) En un RTOS se dispone de 3 procesos (P1, P2 y P3), de los cuales P3 tiene 3 hilos de ejecución (H31, H32 y H33). Describir cómo y en qué momento el scheduler del RTOS decide el proceso e hilo a ejecutar, de qué manera le cede la ejecución, y de qué manera la vuelve a recuperar para decidir un cambio de proceso o hilo.